CINVESTAV-IPN
MEXICO

PROTOCOLO DE ACOTAMIENTO DE DISTANCIAS BASADO EN CANALES CON RUIDO

Valeri Korjik

Sección de Comunicaciones.

CINVESTAV-IPN., MEXICO.

vkorjik@mail.cinvestav.mx

Guillermo Morales-Luna

Sección de Computación.

CINVESTAV-IPN., MEXICO

gmorales@cs.cinvestav.mx

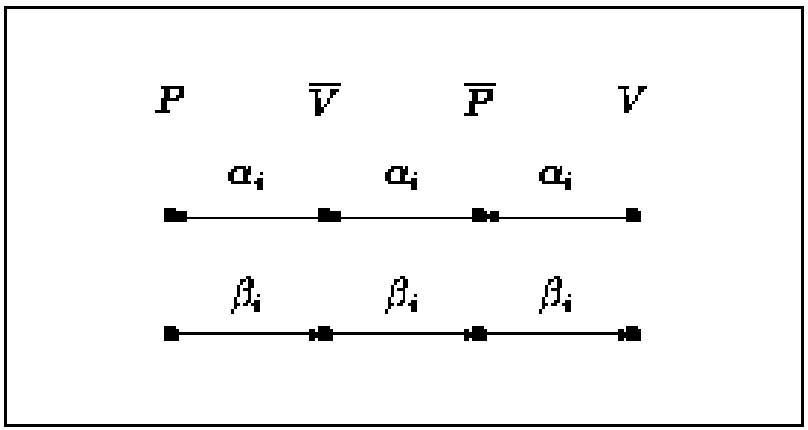
Kirill Morozov

Departamento de Seguridad de Telecomunicaciones

Universidad Estatal de Telecomunicaciones. San Petersburgo, RUSIA

kirill@fem.sut.ru

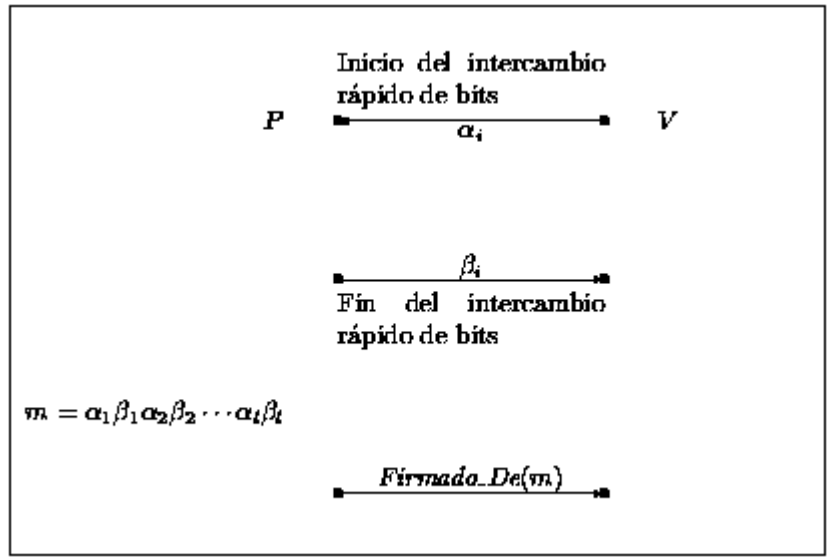
INTRODUCCION



Un *fraude mafioso* en un esquema de identidad

- *Fraude mafioso* : fraude en tiempo real aplicado en esquemas de identificación por un corresponsal fraudulento y un verificador , cooperantes uno con el otro, sin necesidad de conocer porción alguna de información secreta de ellos mismos.
- Cuando *P* está a punto de iniciar el protocolo con $\overline{V}$, $\overline{V}$ establece, digamos, un enlace por radio con $\overline{P}$ para enviarle toda información que le transmita *P*, y a su vez ha de enviársela a *V*. La misma estrategia es aplicada por $\overline{P}$, para enviar a toda información que le haya enviado *V* , para que se la haga llegar a *P* .

No hay métodos prácticos para prevenir fraudes mafiosos salvo por la *técnica de acotamiento de distancias*[1].



A partir de aquí, V puede determinar una cota superior a la distancia a P usando el mayor de los tiempos de retardo entre el envío de cada bit α_i y la recepción del correspondiente bit β_i de vuelta, $i \leq l$.

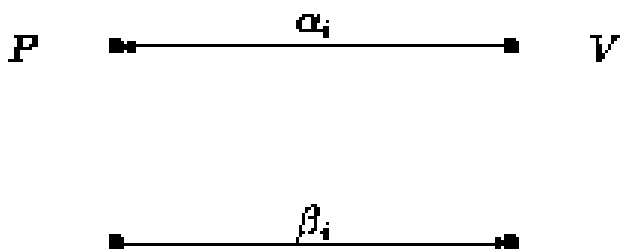
V aceptará este protocolo si y sólo si
P está cerca de V y la firma recibida es la correcta de P
para el mensaje $m = \alpha_1\beta_1\alpha_2\beta_2\cdots\alpha_l\beta_l$.

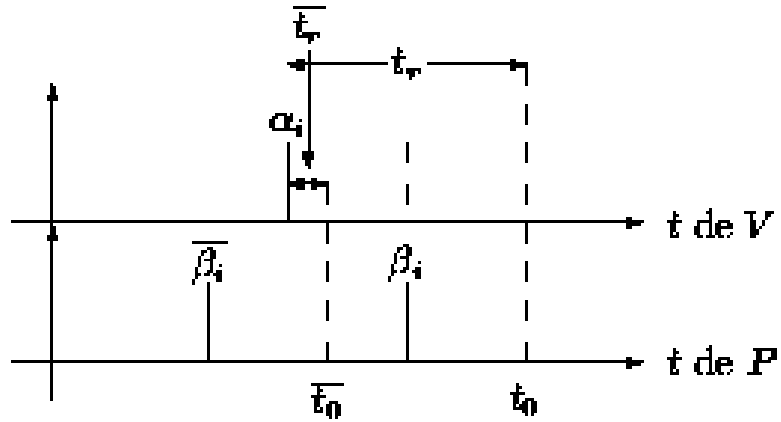
Es claro que si el esquema de firmado es seguro y P no está cerca de V, entonces un fraude mafioso tiene una probabilidad de éxito de a lo sumo 2^{-l}.

Otro ataque: P falsifica distancia

CINVESTAV-IPN
MÉXICO

P , quien posee su llave secreta, es deshonesto y procura falsificar su distancia al verificador

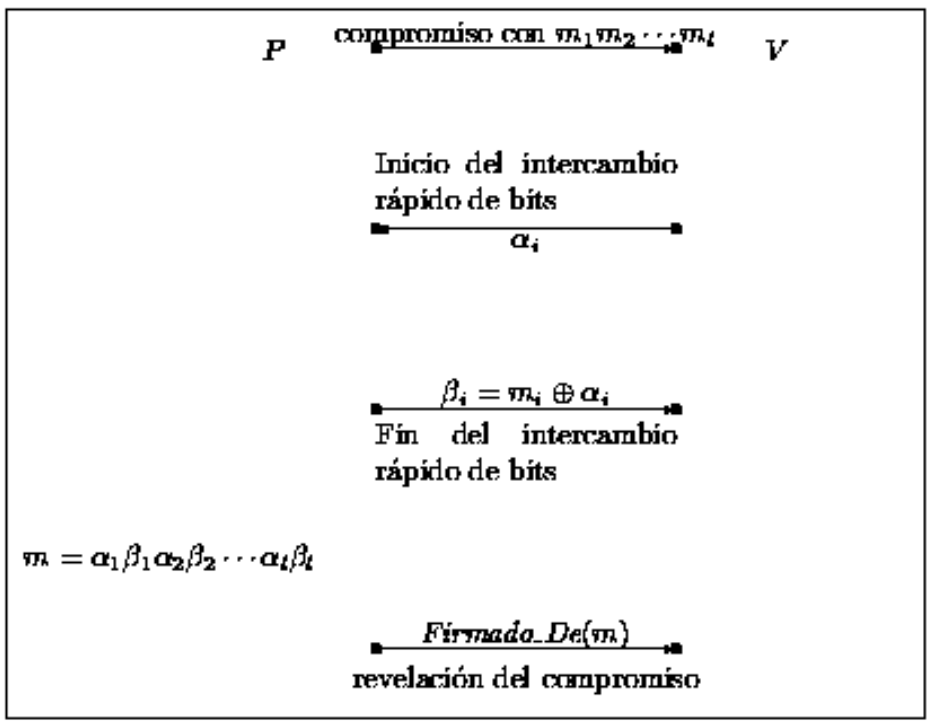




$\overline{t_r} = t_f$ — t_r es el verdadero retraso en tiempo, t_f es un falso retraso en tiempo

Si *P* conociese los tiempos en los que *V* le enviará sus bits α_i , puede proceder a enviarle a *V* sus propios bits $\overline{\beta_i}$ en tiempos "correctos" antes de que arribe α_i , sin importar su distancia a *V* . En consecuencia, el protocolo mostrado NO previene este fraude. Es necesario pues escoger los bits β_i en función de los bits α_i , v. gr., $\beta_i = \alpha_i$ mas así subsiste la vulnerabilidad al ataque de fraudes mafiosos.

Para evitar ambos tipos de fraudes hagamos $\beta_i = \alpha_i \oplus m_i$, donde la secuencia de bits $m_1, m_2, \ldots, m_l$ es generada por P y V queda comprometido por ella, utilizando un *esquema de compromiso* (*commitment scheme*) seguro. Tal protocolo, propuesto en [1], es el siguiente:



Con la información del cuarto paso, V verifica si acaso los bits $\tilde{m}_i = \alpha_i \oplus \beta_i$ coinciden con los bits comprometidos m_i por P . Si así fuera, entonces V calcula m de igual forma a como lo hace P y verifica si la firma recibida corresponde a la de P para m.
En caso afirmativo, V calcula una cota superior de su distancia a P mediante el máximo de los tiempos de retraso y, finalmente,
V habrá de aceptar si y sólo si P está cerca de V .

CINVESTAV-IPN
MÉXICO

PROTOCOLO BC SEGURO COMPUTACIONALMENTE

Un protocolo de compromiso por bits (BC) es muy importante en el protocolo de acotamiento de distancias, previene ambos fraudes: tipo “mafioso” y tipo corresponsal deshonesto.

Protocolo BC: Un corresponsal (Alberto) desea comprometerse con una predicción $m = m_1 m_2 \cdots m_l$ pero no quiere revelar esa predicción a un miembro de la mafia (Marco) sino hasta algún momento posterior. El verificador (Benito), por su lado, desea asegurar que Alberto no cambie de parecer después de que él se haya comprometido con su predicción (pues si se le permitiera hacerlo, si Alberto fuese deshonesto, él podría romper el protocolo de acotamiento de distancias).

El protocolo puede instrumentarse mediante el sencillo algoritmo siguiente:

- Benito genera una cadena aleatoria de bits *R* y se la envía a Alberto.
- Alberto calcula el texto cifrado $c = E_L(R, m)$ y se lo envía a Benito.
  L es una llave aleatoria y la pareja (R, m) denota a la concatenación de *R* con *m*.

Si Alberto decide revelar su compromiso, puede realizar el protocolo siguiente:

- Alberto envía la llave *L* a Benito.
- Benito desencripta el mensaje y recupera (R, m). Habiendo recuperado *m*, al comprobar que aparezca su cadena aleatoria *R* está verificando también la validez de la cadena *m*.

CINVESTAV-IPN
MÉXICO

PROTOCOLO BC BASADO EN CANALES CON RUIDO

Para mantener la seguridad del protocolo BC es necesario imponer condiciones de cómputo limitado a Marco y a Alberto, pues de otra forma cualquiera de Alberto o Marco podría abrir *m* con anticipación y falsificar su distancia, o bien Alberto podría cambiar su compromiso y proceder a falsificar su distancia. Aquí no introduciremos suposiciones de cómputo limitado.

Supongamos que Alberto y Benito se conectan mediante un *canal simétrico binario*, BSC_p, un canal que cambia el valor de cada bit con una probabilidad *p* durante su transmisión de una parte a la otra (supondremos que hay algún otro canal sin ruido conectando ambas partes). Seguiremos el protocolo en [2], extendido a compromisos de cadenas de bits.

Alberto y Benito convienen en utilizar un código binario lineal (n,k), C, con distancia minimal de código, *d*. Sea BSC_p el canal mediante el cual Alberto envía mensajes a Benito.
Similarmente sea $BSC_{p'}$ un canal simétrico entre Alberto y Marco.

Tras de una fase de inicialización, Alberto y Benito realizan el *protocolo 1*:

- Alberto elige una palabra aleatoria $c \in C$, en C, y una cadena aleatoria de n bits, b. Con ello, Alberto calcula una cadena x de l bits tal que

$$m = x \oplus h_b(c) \qquad (1)$$

donde $h_b : C \to \{0,1\}$ es una función de dispersión (*hash*) elegida en la clase $Universal_2$ de [3] y determinada por la cadena b. A cada palabra de código c la transforma en una cadena de l bits, donde l es la longitud de la cadena m con la que se ha de establecer el compromiso.

- Alberto envía c a través del BSC_p y le transmite a Benito, por medio de los canales sin ruido, las palabras b y x.

- Benito conserva c' , b y x, donde c' es la versión recibida de c, acaso corrupta por el canal ruidoso.

Si Alberto quisiese revelar su compromiso *m*, inicia el siguiente *protocolo 2*:

- Alberto envía *c* a Benito, por medio de un canal sin ruido.
- Benito revela la palabra comprometida *m*, calculada mediante la expresión (1), pues él conoce *x*, *b* y *c*. Calcula la distancia de Hamming $d_H(c,c')$ entre *c* y c', y si acaso $d_H(c,c') \le l_0$, donde l_0 es un cierto umbral, entonces Benito acepta *m*. En otro caso la rechaza.

Factores relevantes en la evaluación del protocolo:

- La probabilidad P_c de una *decodificación óptima correcta* de *m* por Marco, basada en su conocimiento de *C*, c'', *b*, *x* y la función de dispersión h_b, donde c'' es la versión que él recibió de *c*, corrupta por el canal $BSC_{p'}$ entre Alberto y Marco. La probabilidad P_c queda acotada superiormente mediante la desigualdad de Fano

$$l + (1-P_c)\log_2 \frac{1-P_c}{2^l - 1} + P_c \log_2 P_c \le I_0 \qquad (2)$$

donde I_0 es la cantidad de información de Shannon que le falta a Benito (y a Marco) sobre *m* dados c'', *b*, *x* y la función de dispersión h_b. El valor I_0 puede a su vez ser acotado mediante el Teorema de Ampliación de la Privacidad [4]:

$$I_0 \leq \frac{2^{-(n-r-t-l)}}{\ln 2} \qquad (3)$$

y aquí

$r = n - k$ es el número de símbolos a revisar en el código *C*, y

$t = n\left(1 + \log_2\left(p'^2 + (1-p')^2\right)\right)$ es la información de Renyi [4] que Marco gana con c'', recibido a través del canal $BSC_{p'}$.

- La probabilidad P_{FR} de que ocurra $d_H(c,c') > l_0$. En esta situación, la palabra comprometida *m* es *falsamente rechazada* por Benito, a pesar de que Alberto ha sido honesto en su ejecución del protocolo. Se puede ver fácilmente que P_{FR} puede acotarse como sigue:

$$P_{FR} \leq \sum_{i=l_0+1}^{n} \binom{n}{i} p^i (1-p)^{n-i} \qquad (4)$$

- La probabilidad P_{FD} de que se elija $\tilde{c} \in C$, $\tilde{c} \neq c$ tal que $d_H(\tilde{c},c') \leq l_0$. Eligiendo una tal $\tilde{c}$, Alberto puede tener éxito en un intento por cambiar su compromiso y consecuentemente en *falsificar su distancia* a Benito.

Suponiendo que un Alberto deshonesto envía una cadena *w* tal que la palabra más cercana $c \in C$ en el código *C* ocurre a una distancia de Hamming *s* a *w*, y tomando en cuenta que para cualquier otra palabra $\tilde{c} \neq c$ en el código se debe cumplir $d_H(w, \tilde{c}) \geq d - s$, como una consecuencia de la propiedad de la distancia de Hamming podemos estimar la probabilidad de que anuncios de cualquier otra palabra $\tilde{c} \in C$ en el código que haya sido enviada por Alberto sea también aceptada por Benito:

$$P_s(\tilde{c}) \leq \sum_{i=0}^{d-s} \binom{d-s}{i} (1-p)^i \, p^{d-s-i} \sum_{j=0}^{l_0-i} \binom{n-d+s}{j} p^j (1-p)^{n-d+s-j} \qquad (5)$$

Por tanto, la probabilidad P_{FD} de que se cambie el compromiso de Alberto, y en consecuencia se falsifique su distancia a Benito, queda acotada superiormente como sigue

$$P_{FD} \leq \max\left\{ \frac{1}{2^l}, \max_{1 \leq s \leq d/2} P_s(\tilde{c}) \right\} \qquad (6)$$

Selección de parámetros principales del protocolo BC utilizado como parte del protocolo de acotamiento de distancias:

- Fíjese los valores de P_{FR}, P_c , P_{FD} , n, p y p'.
- Tómese $l = -\log_2 P_c + 1$.
- Encuéntrese un umbral l_0 de la ecuación (4), dados P_{FR} , p y n.
- Fíjese I_0 que cumpla con la relación (2), dados I y P_c .
- Calcúlese el mínimo $k = n - r$, donde r ha de cumplir con la condición (3).
- Encuéntrese la mínima distancia de Hamming d del código (n,k)-lineal utilizando la cota para códigos BCH, [5].
- Calcúlese P_{FD}' utilizando (6) y revísese si $P_{FD}' \leq P_{FD}$. Si **sí** actualícese $P_{FD} := P_{FD}'$, decreméntese n y repítase el algoritmo para tener un protocolo BC más eficiente. En otro caso, encuéntrese $l > -\log_2 P_c + 1$ que minimice P_{FD}' repitiendo del paso 3. al 7. Si el mínimo P_{FD}' calculado así excediese a P_{FD}, entonces increméntese n y repítase el algoritmo.

Ejemplo: Consideremos $P_c = 10^{-6}$, $P_{FR} = 10^{-4}$, $P_{FD} = 4.5\times10^{-5}$, $l = 20$.
En este caso, Marco (la Mafia) puede falsificar su distancia con probabilidad a lo sumo 10^{-6}. Un Alberto deshonesto tendría una probabilidad de falsificar su distancia a Benito de a lo más 4.5×10^{-5}. De manera simultánea, si ambos Alberto y Benito fuesen honestos y cumplieran con el protocolo con toda precisión, entonces éste se rompería con probabilidad a lo más 10^{-4}
Así, la elección hecha de los parámetros proporciona un protocolo de acotamiento de distancias a la vez seguro y confiable.

Procediendo según el algoritmo, con principales parámetros $n = 16383$, $p = 0.18$ se obtiene

$$l_0 = 3133,\ I_0 = 1.6\times10^{-9},\ r = 8114,\ k = 8164 \text{ y } d = 1175.$$

Esto significa que ambos Marco y Alberto, en un papel deshonesto, prácticamente no tienen posibilidad alguna de falsificar su distancia a Benito.
(Si bien la longitud n del código C es muy grande, éste no se usa para corregir errores sino solamente para proporcionar una distancia mínima de código que sea grande)

CINVESTAV-IPN
MÉXICO

DISCUSION DE LOS RESULTADOS PRINCIPALES

- Observamos que la aplicación del protocolo BC al problema de acotamiento de distancias crea una extraña versión en la que una predicción *m* ha de ser protegida para que no se revele ante la Mafia (no el verificador) y no sea alterada por un eventual corresponsal deshonesto.

- Hemos introducido criterios de evaluación al protocolo de acotamiento de distancias, respecto a seguridad y confiabilidad.

- Hemos obtenido las fórmulas principales, de una manera no-asintótica, para evaluar la seguridad y la confiabilidad del protocolo y optimizar así sus parámetros.

- Hemos presentado un ejemplo para ilustrar el funcionamiento correcto del protocolo. Sin embargo, se hace necesario que el corresponsal envíe una larga cadena de bits al verificador.

CINVESTAV-IPN
MÉXICO

PRINCIPALES PROBLEMAS ABIERTOS EN EL TEMA

- Ver cómo modificar el protocolo BC para garantizar niveles aceptables de seguridad y confiabilidad intercambiando cadenas de bits más cortas entre las partes.

- Ver cómo hacer a todo protocolo de acotamiento de distancias incondicionalmente seguro. (Se requiere de firmas digitales incondicionalmente seguras, cfr.[6])

- Ver cómo establecer BSC_p entre dos partes que proporcione probabilidades óptimas de errores de bits y que no pueda ser modificado por ninguna de las partes deshonestas. (Un enfoque puede ser el establecer BSC_p mediante un canal cuántico. Vale la pena resaltar aquí que el canal cuántico no es necesario para realizar el protocolo BC, sino tan solo para establecer el BSC_p . Contemplamos presentar este enfoque en un futuro próximo.)

CINVESTAV-IPN
MÉXICO

REFERENCIAS

[1] S. BRANDS, D. CHAUM, *Distance-bounding protocols*, Proc. Eurocrypt '93, Lecture Notes in Computer Science, nr. 765, (1993), pp. 344-359.

[2] C. CREPEAU, *Efficient cryptographic protocols based on noisy channels,* Proc. Eurocrypt '97, Lecture Notes in Computer Science, nr. 1233, (1997), pp. 306-317.

[3] D. R. STINSON, *Universal hashing and authentication codes*, Advances in Cryptology, Proc. Crypto '91, Lecture Notes in Computer Science, nr. 576, (1992), pp. 74-85.

[4] C. BENNET, G. BRASSARD, C. CREPEAU, U. MAURER*, Generalized privacy amplificaction*, IEEE Trans. on IT, vol. 41, nr. 6, (1995), pp. 1915-1923.

[5] F. G. MAC WILLIAMS AND N: J. A: SLOAN, *The theory of error-correcting codes*, New York, North-Holland, (1976).

[6] D. CHAUM, S. ROIJAKKERS, *Unconditionally-secure digital signatures,* Proc. Crypto '90, Lecture Notes in Computer Science, nr. , (1991), pp. 206-214